Maintaining Communication Link for a Robot Operating in a Hazardous Environment

Hoa G. Nguyen, Narek Pezeshkian, Anoop Gupta, and Nathan Farrington

Unmanned Systems Branch, Code 2371, Space and Naval Warfare Systems Center, San Diego, CA 92152-7383, hoa.nguyen@navy.mil

Abstract – *We address the problem of maintaining a robust high-bandwidth RF communication link between a mobile robot and its remote control/monitoring station. The solution we are exploring uses a number of autonomous mobile relay nodes. These slave robots convoy behind the teleoperated or autonomous lead robot and automatically stop where needed to maintain an ad hoc network that guarantees a link between the lead robot and its control station. Their mobility allows for more versatility in the network. Nodes that are no longer needed in the network have the ability to navigate back to the lead robot, in order to redeploy at a later time. This further extends the lead robot's range. This paper describes the system, strategy, hardware development, software algorithms, and experiments conducted.*

I. INTRODUCTION

Communication is usually the limiting factor governing human-robot interaction during teleoperated operation in nuclear storage facilities. Thick concrete shielding makes it extremely difficult to maintain high-bandwidth radio communication. [1] The same problem is encountered in urban law-enforcement applications [2] and in hostile military operations. Hard cable tethers are cumbersome, require large spools for extended range, and are not appropriate for most applications besides urban explosive-ordnance-disposal. Thinner optical fibers, even reinforced, have been found to be fragile in field use in the Afghanistan and Iraq theaters. The cables often get run over by the robot as it is maneuvered around obstacles. Snagging and stretching of the fiber around corners often cause signal loss.

These problems may be mitigated if the robot can assume some of the lower-level functions, such as obstacle avoidance and local path planning. This would reduce the amount of data traffic, and allow for the use of lower-frequency, lower-bandwidth radio links that have better wall penetrating capability. However, we have found that in critical missions, the soldier/operators preferred complete control of every aspect of the robot, minimizing any chance for surprises. [3] Thus a high-bandwidth (video rate) communication link is required.

We are investigating the use of radio relays to provide a robust high-bandwidth communication link without the use of cumbersome, fragile, and/or range-limiting cables. Our objectives are to provide a relaying system that functions without distracting the robot operator, significantly extends the robot's range in non-line-of-sight scenarios, and allows for the automatic extraction of the robot and relay nodes after mission completion.

II. APPROACH

To accomplish these goals, we designed a system of mobile relay nodes (essentially slave robots carrying radio relays) that automatically establishes an ad hoc radio network providing an end-to-end link between the robot and its control station. We examined several strategies for network deployment [4], with the main selection criterion being autonomous operation without operator intervention or distraction. The selected deployment strategy calls for the relay nodes to convoy behind the lead robot at the start of each mission (see Fig. 1). Each node monitors the radio link to the node behind it (with the base station being the last node in the system). When the quality of that link drops below a preset threshold, the node stops and becomes a stationary relay node. This is the first half of the project, and is functionally equivalent to the robot carrying a number of relay "bricks" and dropping them off as needed (which is a valid application in itself).

The second half of the project deals with relay redeployment and extraction. As the lead robot maneuvers in a large and complex environment, it may encounter radio-frequency (RF) short cuts that may make a relay node become unnecessary. Each relay node monitors its own usage, and when it detects that it is no longer in the network path between the lead robot and the base station, it will initiate a sequence of steps to allow it to be reused by the system. This begins with a request for a map from the lead robot. Regardless of the mission, an independent subsystem of the lead robot automatically maps the space

Report Documentation Page					Form Approved OMB No. 0704-0188
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE **MAR 2004**		2. REPORT TYPE **N/A**			3. DATES COVERED **-**
4. TITLE AND SUBTITLE **Maintaining Communications Link for a Robot Operation in a Hazardous Environment**					5a. CONTRACT NUMBER
					5b. GRANT NUMBER
					5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Hoa G. /Nguyen; Narek /Pezeshkian; Anoop /Gupta; Nathan /Farrington**					5d. PROJECT NUMBER
					5e. TASK NUMBER
					5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Space and Naval Warfare Systems Center Unmannedd Systems Branch, Code 2371 53406 Woodward Road San Diego, CA 92152-7383**					8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)					10. SPONSOR/MONITOR’S ACRONYM(S)
					11. SPONSOR/MONITOR’S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release, distribution unlimited**					
13. SUPPLEMENTARY NOTES **The original document contains color images.**					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**8**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

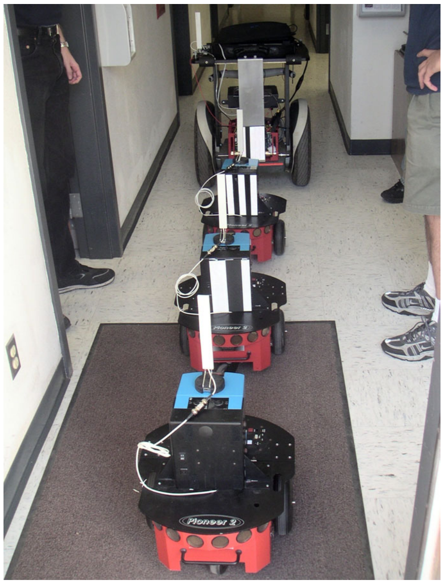

Fig. 1. The convoy at the start of a mission.

it traverses, and passes this map back to any relay robot that requests it. The relay robot will then use the map to seek out and catch up to the lead robot, rejoin the remaining convoy, and redeploy as needed. This will extend the range of the lead robot significantly. The map navigation capability of the relay nodes also means that they can be recalled at the end of the mission.

For laboratory demonstrations, we leveraged our existing pool of laboratory robots. We have used several different robots as the lead robot, including an iRobot ATRV, our own ROBART III [5], and a Segway Robotic Mobile Platform (RMP, see Fig. 2), which was developed by Segway LLC with our coordination. [6]

Fig. 2. The Segway RMP configured as a lead robot.

We are using multiple ActivMedia Pioneer 2-DX robots as relay nodes. These robots are only meant for laboratory demonstrations; all system functionalities can be readily transferred to actual ruggedized field robots.

The Pioneer robots are equipped with 16 Polaroid sonar sensors that we use only for obstacle avoidance. Each robot (including the lead robot) is also equipped with a SICK LMS200 laser radar (ladar). [7] The ladar is used in multiple functions, including obstacle avoidance, beacon identification, mapping, map-based localization, and beacon-based localization.

For closed-loop control of the relay robots, we installed a Compulab 686CORE/686BASE 266MHz Pentium class single-board computer in each. These computers provide enough processing power for the beacon tracking and map navigation functions. (We previously reported using Bright Star Engineering's StrongARM-based nanoEngines. [4] We have subsequently found these computers to be inadequate because some of our current algorithms are floating point intensive and the StrongARM CPU lacks a floating point unit.)

III. AD HOC NETWORKING RADIOS

For uninterrupted operations requiring no operator intervention, we needed a networking scheme that guarantees a solid link between the lead robot and the base station at all times, as the robot moves about in its environment. We worked with BBN Technologies to produce this capability in a small package, using software developed by BBN under a separate DARPA project. [8]

BBN's ad hoc networking software uses a proactive link-state protocol. Each node in the network has complete information about the characteristics of all links. It can execute a routing algorithm of its choice and determine the paths most suitable for the chosen criteria. Each node uses broadcast messages (sent at intervals determined by the network criteria and the environment) to determine the characteristics of the links and set up the routing table, which is recomputed whenever certain network events occur (such as when the link quality between two nodes has dropped below a preset level appropriate for a desired scenario). The routing table can thus be updated before a link is broken, and the network is automatically maintained in a proactive fashion, for optimal information transmission and minimal lag. There is no delay incurred for route re-selection due to broken links.

We integrated this software into a small radio only slightly larger than a pack of playing cards. Each radio consists of a nanoEngine processor card, an off-the-shelf

802.11b PC Card, and a radio interconnect board (RIB) that interfaces the two components and provides power conditioning, Ethernet, and serial ports. [4] Six prototypes were originally developed for our project. Later, 100 more were produced for use on other DARPA robotics research projects at University of Pennsylvania, Georgia Institute of Technology, University of Southern California, and BBN Technologies, as well as other in-house robotics projects.

We subsequently developed a second-generation RIB that provides two PC Card slots and a USB Host controller onboard (see Fig. 3). The additional PC Card slot could be used to house a second 802.11 radio to provide higher speed relaying (through the simultaneous use of two radios on different channels). The USB port allows the use of inexpensive web cameras on the relay nodes to provide a rearguard function. (Without the USB port, more expensive Ethernet cameras or a combination of analog video cameras and CODEC boards, such as the Indigo Vision VP604, could still be used.)

Fig. 3a. One side of the RIB2, showing the nanoEngine.

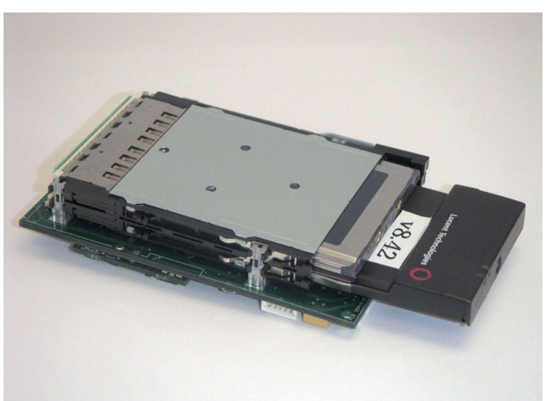

Fig. 3b. The other side of the RIB2, showing two PC Cards

IV. CONVOYING USING RETROREFLECTIVE BEACONS

Our convoying algorithm (a copy of which runs on each relay robot) uses the Pioneer's ladar and a set of tools developed by the University of Southern California, namely the Stage simulator and the Player robot device server. [9] Each robot has a beacon attached to its back (see Fig. 4). The beacon itself is a 5-bit barcode formed from strips of retroreflective tape. The tape appears brighter to the ladar than everything else in the environment. Player provides a component that allows the ladar to obtain the range, bearing, orientation, and unique identification of the other robots' retroreflective beacons. Each relay robot uses this information to form a convoy, with steering and velocity commands based on the bearing and range to the desired beacon.

Fig. 4. A relay node with retroreflective barcode beacon. This barcode is the number 21 (10101 in binary).

While convoying, each relay robot also needs to perform obstacle avoidance (mainly to negotiate around corners without running into them). The robots use both ladars and sonars to detect obstacles. Each type of sensor has its own limitations. Sonars have a minimum distance below which they cannot detect objects. Because of their relatively low frequency, they also suffer from specular reflections off flat surfaces. [10] (At certain angles, specular reflections return incorrect ranges, or often do not return at all). Ladar suffers from specular reflections off of black and shiny surfaces (such as black metal desks). Also, 2D ladar only works on a flat plane parallel to the floor, while the sonar beam pattern is conical. A

fusion of the two sensor types compensates for the limitations of each. Fig. 5 shows superimposed sonar and ladar scans. The sonar return at the bottom left came from an open door (the ladar only covers the front 180 degrees). The sonar and ladar returns at the top left came from specular reflections off of a black, shiny metal desk.

We fused the results from the two sensor types by dividing the 120-degree arc in front of the robot into six 20-degree "obstacle zones." There are 6 sonar sensors, one for each 20-degree zone. Any sonar or ladar return inside a zone that is shorter than a predetermined obstacle distance places a "1" into that zone. To eliminate occasional false positives from the ladar, which has a very fine 0.5-degree angular resolution, we require two such returns from the ladar in each sector before it is registered.

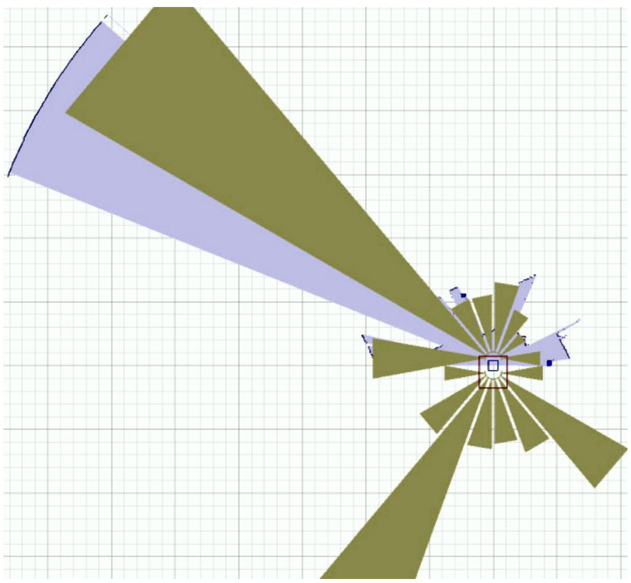

Fig. 5. Superimposed sonar (dark) and ladar (light) data from a Pioneer relay robot. The image was captured using USC's Player Viewer tool.

The obstacle zone vector then determines whether a steering command from the laser beacon algorithm (issued at a 5 Hz rate) is passed through unmodified, or is modified or replaced by an obstacle avoidance command in a reactive, subsumption-like manner (see Table I). Left and right turn obstacle-avoidance commands are executed with forward velocity in tact, while turn-in-place commands are executed with zero forward velocity and with direction being that of the previous turn direction (to prevent oscillations). This obstacle avoidance behavior, while extremely simple, proved quite adequate for our convoying task. The convoying behavior was thus developed on the Stage robot simulator [4], and then transferred to our Pioneer robots for real-world demonstrations.

TABLE I. Sample obstacle avoidance behaviors.

Obstacle Zone Vector						**Steering Modification**
0	0	0	0	0	0	None
0	0	0	0	0	1	Turn left
0	0	0	0	1	0	Turn left
						…
0	0	0	1	0	0	Turn in place
						…
1	1	0	0	0	0	Turn right
						…
1	1	1	1	1	1	Turn in place

V. RELAY DEPLOYMENT EXPERIMENTS

We present here the results of two relay-deployment experiments conducted at our facilities. The first experiment was performed in a mixed indoor/outdoor environment, while the second involved traversing an underground bunker.

V.A. Mixed Indoor/Outdoor Environment Experiment

Fig. 6 shows the path taken by a teleoperated lead robot (a Segway RMP) and the relay convoy, and the final locations of the deployed relay nodes. Node 1 is the lead robot, and node 5 is the base station (one of the radios connected to a laptop via an Ethernet cable), located in the first author's office. The convoy started in the hallway outside the office, as shown in Fig. 1. Node 4 (the last Pioneer robot in the convoy) stopped just after the turn into the open laboratory area, as the link quality between it and node 5 dropped to a preset level. Nodes 3 and 2 likewise stopped in the open courtyard between the buildings. In each case, the relay node stopped just after line of sight is lost to the node behind it, consistent with the expectation that high-bandwidth digital RF links operate mostly on lines of sight (an assumption we made at the start of the project [4]).

V.B. Underground Bunker Experiment

The objective of our second experiment was to teleoperate a mobile robot (again, a Segway RMP) through an underground bunker with thick concrete walls, from one side of a hill to the other. This would approximate the environments encountered in tunnel and cave explorations, as well as inspection of underground nuclear storage facilities.

The operator's control station (node 5) was stationed outside the southwest entrance of Battery Woodward (an

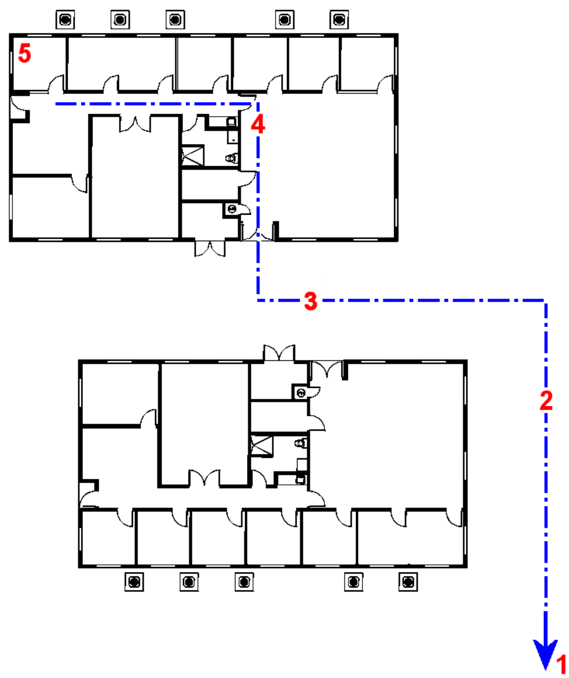

Fig. 6. Mixed indoor/outdoor relay deployment experiment.

abandoned World War II gun battery and underground bunker protecting the coast of San Diego, see Fig. 7). As in the previous experiment, each relay node stopped just after line of sight to the node behind it was lost. The experiment stopped after a high iron door threshold blocked node 2's advance. Nevertheless, the lead robot had made it through to the east entrance of the bunker, from one side of the hill to the other, operated solely by real-time video relayed through the network.

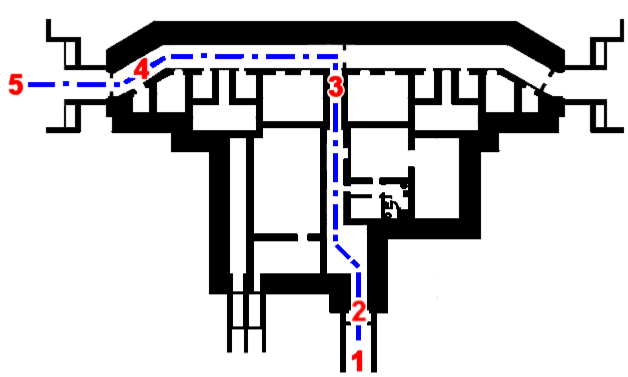

Fig. 7. Underground bunker relay deployment experiment.

VI. RELAY REUSE

In order for a relay robot to catch up to the convoy, the relay robot must (1) know where it is, (2) know where the lead robot and the remaining convoy are, and (3) know how to navigate to the convoy's location. If the convoy has moved out of sight of the stationary relay robot, then problems 1 and 2 require that all robots localize themselves to the same reference frame for navigation coordinates to be valid between robots. Problem 3 requires that the relay robot possess a map of the environment so that it can plan a collision-free path from its current location to the convoy. The lead robot generates this map as it moves through the environment, and sends it back to the relay robot when requested.

VI.A. Localization

To solve problem 1, we have selected the Adaptive Monte-Carlo Localization (AMCL) algorithm [11] as implemented in the Player software toolset. This algorithm matches measurements from the ladar sensor to the map in order to determine the pose (position and orientation) of the robot with respect to the map (see Fig. 8). Since AMCL works best when it has a reasonably good initial guess of the correct pose, we have designed another algorithm that allows the robots in the convoy to continuously maintain their approximate pose information. When a relay robot stops and becomes a static relay node, its final pose estimate from this convoy localization algorithm is stored in the map being generated by the lead robot, and will become the initial pose estimate for the AMCL algorithm when the map is requested by the relay robot at a later time.

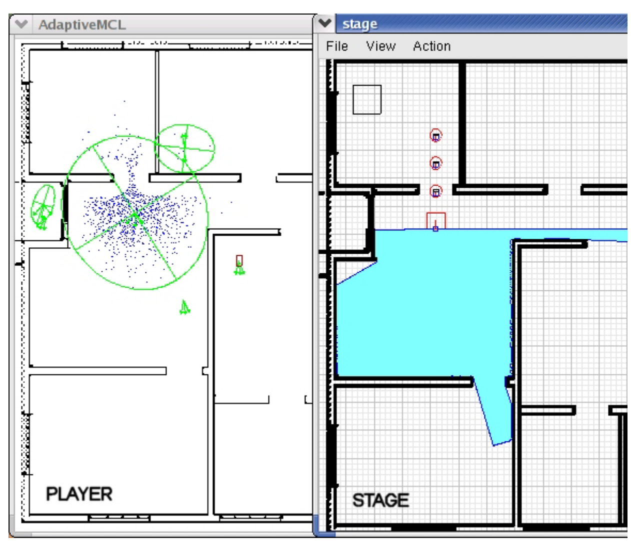

Fig. 8. AMCL pose estimates (left) and robot location during simulation (right).

Our convoy localization algorithm is a form of beacon-based localization that uses other robots as beacons. The first relay robot in the convoy localizes itself to the lead robot's reference frame. The second relay robot in the convoy localizes itself to the first relay robot, and hence the lead robot's reference frame. This process is applied to all of the relay robots in the convoy and results in all robots sharing the lead robot's reference frame (see Fig. 9).

The reference robot transmits its pose periodically to the next robot down the line using the same wireless communications network. The observing robot obtains the range, bearing, and orientation of the reference robot by using the ladar to measure the fixed-size retro-reflective barcode mounted on the back of the reference robot. Combining these data with the reference robot's pose, the pose of the observing robot can be calculated with respect to the reference robot. This approach is similar to [12].

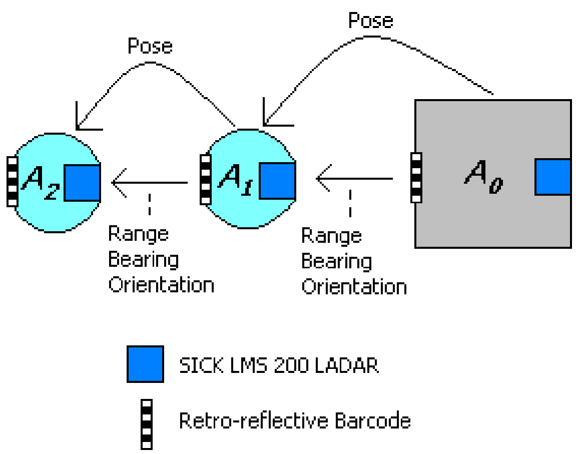

Fig. 9. Convoy localization.

Let us assume that the convoy consists of a lead robot, A_0 (with pose x_0, y_0, and θ_0), and a relay robot, A_1 (with pose x_1, y_1, and θ_1)--see Fig. 10. The Player component that detects beacons returns the range, r, to the beacon, the bearing, b, to the beacon, and the orientation, o, of the beacon. The range is the distance between the ladar and the beacon. The bearing is the angular distance that the ladar would have to turn in order to be directly facing the beacon. The orientation is the angular distance the ladar would have to turn in order for both the ladar and the beacon to be facing the same direction. All angles are negative for clockwise measurements and positive for counterclockwise measurements.

By definition:

$$\theta_1 = \theta_0 - o \tag{1}$$

Given $A_0 (x_0, y_0, \theta_0)$ and $r, b, o,$ let

$$b' = \theta_0 + b - o \tag{2}$$

then

$$x_1 = x_0 - r\cos(b') \tag{3}$$

$$y_1 = y_0 - r\sin(b'). \tag{4}$$

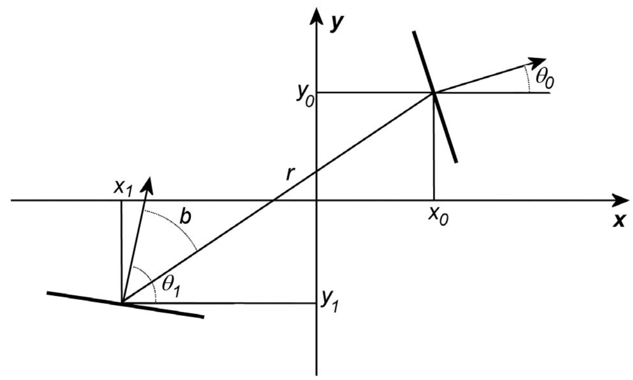

Fig.10. Diagram for deriving the equations.

The above equations assume that the reference center of the ladar and the mid-point of the retroreflective beacon are co-located on each robot. In practice this is not true, and the offset must be taken into account. The calculated results are adjusted before being passed on to the next robot.

This scheme contains several sources of error. Besides the unavoidable inaccuracies in the ladar measurements, a possible error arises from the time difference between the data collecting events at each ladar. The pose of robot A_0 is calculated at time t_0 but transmitted at time t_1 and processed by robot A_1 at time t_2. A_1's ladar also detects A_0's beacon at time t_3. Increasing the sampling and data transmission frequencies can minimize this error. Synchronizing the clocks on both robots and adding time stamps to the data can also help minimize the error.

As the number of relay nodes in the convoy grows, so does the accumulated error in reference frames between the lead robot and the last node in the convoy. For our application, however, we do not require very accurate localization. Our goal is only to gain an estimate of the pose to provide a seed for the AMCL algorithm.

VI.B. Mapping

Problem 2 is more difficult because the lead robot does not possess an a priori map of the environment. Due to wheel slippage and inaccuracies in dead reckoning techniques, the map that is generated is often distorted.

The lead robot may not know its exact position on the map that it generated. A better map can be generated using a technique called Simultaneous Localization and Mapping (SLAM). [13, 14, 15] The lead robot localizes itself to the map that it creates in real time. We are considering using the algorithm by Thrun [13], distributed as part of the CARMEN open-source software package. This algorithm combines an incremental maximum likelihood estimator with a posterior pose estimator to incorporate new ladar data into a map and to maintain consistency with older data, closing cycles in the map.

To communicate the pose of the lead robot and the lead robot's map with the other relay nodes, we have created a blackboard system [16] that allows all of the robots to share information. Each robot maintains its own copy of the blackboard and all blackboard variables. Every time a blackboard variable is modified (such as the pose), a message is broadcast over the ad-hoc network radios to all of the other robots instructing them to update their blackboards as well.

VI.C. Navigation

Once the relay node has localized itself with respect to the map and knows where the convoy is on the map, the last problem to solve is how to navigate from the robot's position to the convoy's position, without colliding with any obstacles.

We are experimenting with a very simple navigation scheme. In this scheme, the lead robot marks the map periodically with its current pose. These can be thought of as virtual breadcrumbs. When the relay robot gets a copy of the map, it also gets a sequence of breadcrumbs that the lead robot has generated. By navigating from one breadcrumb to the next, the relay robot will retrace the path of the convoy and will eventually catch up to the convoy.

To ensure the relay robot's safe autonomous journey back to the convoy, more sophisticated obstacle avoidance may be required. We are investigating the use of the Vector Field Histogram [17] obstacle avoidance algorithm. This algorithm creates a local map and uses it to navigate around obstacles. We have demonstrated this obstacle avoidance algorithm in simulation with the Stage simulator.

VII. CURRENT STATUS

We have successfully demonstrated automatic relay deployment in real-world environments. For the second phase of the project (relay reuse), we have successfully implemented blackboard communication, map-based localization, VFH obstacle avoidance, and beacon-based localization in simulation, but not yet on actual robots. SLAM (using CARMEN) is being installed and demonstrated on ROBART III as part of the DARPA/JRP Technology Transfer project. [18] Bread-crumb navigation still remains to be implemented. Once we integrate these two technologies, we should have a fully functional system capable of relay reuse and robot recall at the end of the mission.

VIII. CONCLUSIONS

Robots have already started to replace humans in dangerous tasks like inspection of nuclear facilities, explosive ordinance disposal, search and rescue, firefighting, mine detection, border patrol, and military surveillance. The most common problem in all of these scenarios is the lack of a robust, high-quality, long-range communications link.

Our approach to the communications problem is to use a dynamic network of short-range, high-throughput digital radios. This high-bandwidth network allows real-time video streaming from the robot to the operator. By putting the network nodes on autonomous robots, the lead robot is not constrained to a preplanned path or environment. Nor must the operator divide attention between the lead robot and the communications relay system. However, making the autonomous relay robots smart enough to survive without a human operator is a difficult challenge.

We have successfully demonstrated the system, with four autonomous mobile relay nodes, in mixed indoor/outdoor environments as well as through underground bunkers with thick concrete walls. The relay nodes were able to convoy behind a lead robot and stop where needed to maintain a high-bandwidth digital video link, significantly increasing the lead robot's range.

At this point, the system is useful for advance reconnaissance and clearing missions, where humans can follow and retrieve the robots after the area has been verified free of hostile forces by the robots. However, for a truly flexible system capable of operating in permanently hazardous environments, two additional capabilities are needed: the ability for the relay nodes to rejoin the lead robot when no longer required in the network, and the ability to be recalled when the mission is completed. We are currently working on these issues.

ACKNOWLEDGMENTS

This project has been funded by the Defense Advanced Research Projects Agency / Information Processing Technology Office (DARPA/IPTO) under the Mobile Autonomous Robot Software (MARS) program.

REFERENCES

1. D. BRUEMMER et al., “Intelligent Autonomy for Remote Characterization of Hazardous Environments”, *IEEE Int. Symp. on Intelligent Control (ISIC),* Houston, TX, 5-8 October 2003.

2. H.G. NGUYEN and J.P. BOTT, “Robotics for Law Enforcement: Applications Beyond Explosive Ordnance Disposal”, *SPIE Proc. 4232: Technologies for Law Enforcement,* Boston, MA, 5-8 November 2000.

3. R.T. LAIRD et al., “Issues in Vehicle Teleoperation for Tunnel and Sewer Reconnaissance”, Workshop 7: Vehicle Teleoperation Interfaces, *IEEE Int. Conf. on Robotics and Automation* (ICRA), San Francisco, CA, 28 April 2000.

4. H.G. NGUYEN et al., “Autonomous Communication Relays for Tactical Robots”, *11th Int. Conf. on Advanced Robotics (ICAR),* Coimbra, Portugal, 30 June - 3 July 2003.

5. G.A. GILBREATH, D.A. CICCIMARO, and H.R. EVERETT, “An Advanced Telereflexive Tactical Response Robot”, Proceedings, Workshop 7: Vehicle Teleoperation Interfaces, IEEE International Conference on Robotics and Automation (ICRA), San Francisco, CA, 28 April 2000.

6. SEGWAY RMP project, Space and Naval Warfare Systems Center, San Diego, URL: h*ttp://www.spawar.navy.mil/robots/land/SegwayRMP/SegwayRMP.html*

7. C. YE and J. BORENSTEIN, “Characterization of a 2-D Laser Scanner for Mobile Robot Obstacle Negotiation”, *IEEE Int. Conf. on Robotics and Automation*, Washington DC, USA, 11 - 15 May 2002, pp. 2512-2518.

8. J. REDI and B. WELSH, “Energy Conservation for Tactical Robot Networks”, *IEEE Military Communications Conf. (MILCOM)*, 31 Oct. - 3 Nov. 1999, pp. 1429-1433.

9. B. GERKEY, R.T. VAUGHAN, and A. HOWARD, “The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems”, *11th Int. Conf. on Advanced Robotics (ICAR)*, Coimbra, Portugal, 30 June – 3 July 2003

10. H. R. EVERETT, *Sensors for Mobile Robots: Theory and Application,* Wellesley, MA: AK Peters, 1995.

11. D. FOX, “KLD-Sampling: Adaptive Particle Filters”, *Advances in Neural Information Processing Systems (NIPS)*, Vancouver, Canada, 3-8 December 2001.

12. A. HOWARD, M. MATARIĆ, and G. SUKHATME, “Localization for Mobile Robot Teams Using Maximum Likelihood Estimation”, *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland, 30 Sep.30 – 4 Oct. 2002.

13. S. THRUN, “A Probabilistic Online Mapping Algorithm for Teams of Mobile Robots”, *Intl. Journal of Robotics Research,* 20(5), pp. 335-363, 2001.

14. J.-S. GUTMANN and K. KONOLIGE, “Incremental Mapping of Large Cyclic Environments”, *IEEE Int. Symp. on Computational Intelligence in Robotics and Automation (CIRA),* Monterey, CA, 8-9 Nov. 1999.

15. B. YAMAUCHI, A. SCHULTZ, and W. ADAMS, “Mobile Robot Exploration and Map-Building with Continuous Localization”, *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Leuven, Belgium, 1998.

16. S. HARMON, W. AVILES, and D. GAGE, “A Technique for Coordinating Autonomous Robots”, *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, San Francisco, CA, 7-10 April 1986.

17. I. ULRICH and J. BORENSTEIN, “VFH+: Reliable Obstacle Avoidance for Fast Mobile Robots”, *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Leuven, Belgium, 1998.

18. TECHNOLOGY TRANSFER project, Space and Naval Warfare Systems Center, San Diego, URL: *http://www.spawar.navy.mil/robots/research/TechXfer/TechXfer.html*